

リカレントニューラルネットワークの複雑さの制御に関する 一考察

Study on Complexity Control Methods for Recurrent Neural Networks

○酒井正夫*, 本間経康**, 阿部健一*

○Sakai Masao*, Noriyasu Honma** and Kenichi Abe*

*東北大学大学院工学研究科, **東北大学医療技術短期大学部

* Graduate School of Engineering, Tohoku University

** College of Medical Sciences, Tohoku University

キーワード: リカレントニューラルネットワーク (recurrent neural networks), カオス (chaos),
リアプノフ指数 (Lyapunov exponents)

連絡先: 〒980-8579 仙台市青葉区荒巻字青葉05 東北大学 大学院 工学研究科 電気・通信工学専攻 阿部研究室
酒井正夫, Tel.: (022)217-7074, Fax.: (022)263-9290, E-mail: sakai@abe.ecei.tohoku.ac.jp

1. はじめに

ニューラルネットワークとは、ニューロンと呼ばれる単純な入出力特性をもつ素子の組み合わせにより生物の脳を模した人工的な情報処理システムである。1986年にRumelhartによって誤差逆伝搬 (Back Propagation: BP) 学習法が提案され、階層型ニューラルネットワークに人間が行うようなパターン認識や運動制御、および非線形モデルによるシステム同定の実現が可能となった²⁾。

しかし、従来のネットワークでは実現できない、より高精度な同定、およびより生物的で柔軟な認識機構を実現するためには、代表的な誤差逆伝搬学習法では考慮していないネットワークダイナミクス³⁾の複雑さを制御する事が重要である事が、動物実験²⁾および計算機シミュレーション⁵⁾, ⁴⁾などの結果より報告されている。

著者らはこれまでに、フィードバック結合を持ち複雑なネットワークダイナミクスが実現可能であるリカレントニューラルネットワーク (Recurrent Neural Networks: 以下RNNと略す) を対象とした複雑さ制御 (Complexity Control: CC) 法のプロトタイプを提案した⁵⁾。本稿では、その提案法の制御性能を計算機シミュレーションの結果を基に詳しく考察し、より制御能力の高い手法を提案する。

2. Recurrent Neural Networks

本研究では、様々な相互作用 (多様な複雑さのダイナミクス) を表現可能なニューラルネットワークとして、各ニューロンの写像関数がシグモイド関数で次式で時間発展していく、離散時間系完全結合型RNNを用いる。

$$x_i(t+1) = \frac{1}{1 + \exp(-2s_i(t+1)/s_0)}, \quad (1)$$

$$s_i(t+1) = \sum_{j=1}^N w_{ij}y_j(t) + \theta_i, \quad (2)$$

$$y_i(t) = x_i(t), \quad (3)$$

ただし、 t は離散時間($t=1, 2, \dots$)で、 N はニューロン数、 s_i , y_i , θ_i , s_0 はニューロンの内部ポテンシャル、出力、閾値、シグモイド関数の定数であり、 w_{ij} はニューロン j からニューロン i への結合係数である。

これらのパラメータ w_{ij} , θ_i などを変化させることによって、RNNは、無秩序でカオス的な振舞いから規則的で単純な振舞いまで様々なダイナミクスを示す³⁾。

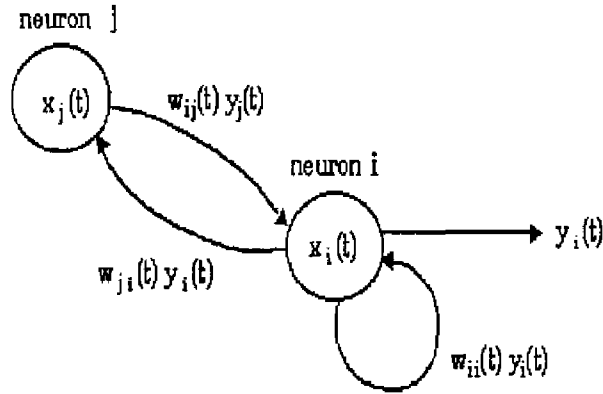


Fig. 1 Fully connected recurrent neural networks

3. 複雑さの定義

本稿ではダイナミクスの複雑さの定量的指標として、最大リアブノフ指数を用いる。一般に、最大リアブノフ指数の値が負の場合ネットワークダイナミクスは規則的で単純であり、正の場合カオス的で複雑ある¹⁾。

n 階差分方程式

$$\mathbf{x}(t+1) = \mathbf{F}(\mathbf{x}(t)), \quad (4)$$

$$\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T, \quad (5)$$

$$\mathbf{F}(\mathbf{x}(t)) = [f_1(\mathbf{x}(t)), f_2(\mathbf{x}(t)), \dots, f_n(\mathbf{x}(t))]^T \quad (6)$$

の最大リアブノフ指数は次式で計算される。

$$\lambda = \lim_{T \rightarrow \infty} \frac{1}{T} \log \left[\frac{\|\mathbf{L}(T)\|}{\|\mathbf{L}(0)\|} \right], \quad (7)$$

ただし $\mathbf{L}(0)$ は $\mathbf{x}(0)$ を中心とした無限小の n 次元球体の直径ベクトルであり、 $\mathbf{L}(t)$ は、その球体を t 回遷移したときに得られる $\mathbf{x}(t)$ を中心とした楕円体の最大の主軸ベクトルである⁶⁾。

RNNの最大リアブノフ指数は、RNNの様々なパラメータに依存する値である。本研究では簡単のため、結合係数 w_{ij} 以外のパラメータを固定と考える。 $\mathbf{x}(t)$ におけるヤコビ行列を $D\mathbf{F}(\mathbf{x}(t))$ とおくと、アトラクタ近傍で線形化して、

$$\mathbf{L}(t+1) = D\mathbf{F}(\mathbf{x}(t))\mathbf{L}(t) \quad (8)$$

となるから、次式を得る。

$$\lambda = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T \log \left[\frac{\|D\mathbf{F}(\mathbf{x}(t))\mathbf{L}(t)\|}{\|\mathbf{L}(t)\|} \right]. \quad (9)$$

ここで、分子の計算が発散しないように、 $\mathbf{L}(t+1)$ を正規化して

$$\mathbf{L}(t+1) = D\mathbf{F}(\mathbf{x}(t))\delta\mathbf{L}(t), \quad (10)$$

とする。ただし、 $L(t) = \|\mathbf{L}(t)\|$ として、

$$\delta\mathbf{L}(t) = \frac{\mathbf{L}(t)}{L(t)}, \quad (11)$$

である。最大リアブノフ指数を求める場合には、 $\delta\mathbf{L}(0)$ は任意方向の n 次元ベクトルで良いから⁶⁾、次式が得られる。

$$\lambda = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T \log L(t). \quad (12)$$

(1) ~ (3)のRNNの場合、ヤコビ行列の (i, j) 要素 $DF_{ij}(\mathbf{x}(t))$ は、次式で表わされる。

$$DF_{ij}(\mathbf{x}(t)) = \frac{2}{s_0} w_{ij} x_i(t+1)(1 - x_i(t+1)). \quad (13)$$

したがって、 $\mathbf{L}(t) = [L_1(t), L_2(t), \dots, L_N(t)]^T$ の要素は、次式で表わされる。

$$L_i(t+1) = \frac{2}{s_0} X_i(t+1) \sum_{j=1}^N w_{ij} \delta L_j(t), \quad (14)$$

また,

$$L(t) = \sqrt{\sum_{i=1}^N (L_i(t))^2}, \quad (15)$$

である. ただし, $\delta L(t) = [\delta L_1(t), \delta L_2(t), \dots, \delta L_N(t)]^T$ とし, $X_i(t) = x_i(t)(1 - x_i(t))$ とする.

以上より, RNNの出力時系列 $\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(T)$ はRNNのニューロンの初期状態 $\mathbf{x}(0)$ が決まればRNNの結合係数 w_{ij} により計算できることから, 任意の $\|\delta L(0)\| = 1$ となる n 次元ベクトルと $\mathbf{x}(0)$ を準備すれば, (14)式の最大リアプノフ指数は, RNNの結合係数 w_{ij} により計算できる. すなわち

$$\lambda = F_\lambda(\mathbf{W}, \mathbf{x}(0)). \quad (16)$$

ただし

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & w_{nn} \end{bmatrix}. \quad (17)$$

4. 複雑さ制御法

前節で示したようにRNNの複雑さの定量的指標である最大リアプノフ指数は, 結合係数 w_{ij} と初期状態 $\mathbf{x}(0)$ によって決定される. したがって, 初期状態 $\mathbf{x}(0)$ を固定した場合, ある複雑さ λ をもつRNNを, 目標とする複雑さ λ^{obj} に設定する問題は, 結合係数 w_{ij} を適当な値 w_{ij}^{obj} に改変する問題に帰着される.

4.1 提案法

著者らが提案した複雑さ制御法は, ネットワークパラメータを適当な値で初期化して, 最急降下法を用いて適切な改変量 δw_{ij} 得る方法である. すなわち, RNNの複雑さが λ , 目標値が λ^{obj} の場合結合係数 w_{ij} の改変量 δw_{ij} は, 目標値との二乗誤差を $e_\lambda = (\lambda - \lambda^{obj})^2/2$ として

$$\delta w_{ij} = -\eta \frac{\partial e_\lambda}{\partial w_{ij}},$$

$$= -\eta(\lambda - \lambda^{obj}) \frac{\partial \lambda}{\partial w_{ij}}. \quad (18)$$

で与えられる. ここで, $\eta(> 0)$ は学習係数である.

$\partial \lambda / \partial w_{ij}$ は式(12)より,

$$\frac{\partial \lambda}{\partial w_{ij}} = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \frac{1}{L(t)} \frac{\partial L(t)}{\partial w_{ij}}, \quad (19)$$

であり, $\partial L(t) / \partial w_{ij}$ は(15)式より

$$\frac{\partial L(t)}{\partial w_{ij}} = \frac{1}{L(t)} \sum_{i=1}^N L_i(t) \frac{\partial L_i(t)}{\partial w_{ij}}, \quad (20)$$

である. また $\partial L_i(t) / \partial w_{ij}$ は, Δ_{ik} をクロネッカーのデルタとして

$$\begin{aligned} \frac{\partial L_k(t+1)}{\partial w_{ij}} &= \frac{2}{s_0} \left(\frac{\partial X_k(t+1)}{\partial w_{ij}} \sum_{m=1}^N w_{km} \delta L_m(t) \right. \\ &\quad \left. + X_k(t+1) \left[\Delta_{ik} \delta L_j(t) \right. \right. \\ &\quad \left. \left. + \sum_{m=1}^N w_{km} \frac{\partial \delta L_m(t)}{\partial w_{ij}} \right] \right). \end{aligned} \quad (21)$$

ここで, $\partial X_k(t+1) / \partial w_{ij}$ は,

$$\frac{\partial X_k(t)}{\partial w_{ij}} = (1 - 2x_k(t)) \frac{\partial x_k(t)}{\partial w_{ij}} \quad (22)$$

となり, $\partial x_k(t) / \partial w_{ij}$ は, 式(1)~(3)より

$$\begin{aligned} \frac{\partial x_k(t)}{\partial w_{ij}} &= \frac{\partial x_k(t)}{\partial s_k(t)} \frac{\partial s_k(t)}{\partial w_{ij}} \\ &= f'(s_k(t)) (\Delta_{ik} \cdot x_j(t-1) \\ &\quad + \sum_{m=1}^N w_{km} \frac{\partial x_m(t-1)}{\partial w_{ij}}) \end{aligned} \quad (23)$$

で求まり, $\mathbf{x}(0)$ は結合係数 w_{ij} に依存しないので $\partial x_k(0) / \partial w_{ij} = 0$, $\partial x_k(1) / \partial w_{ij} = \Delta_{ki} f'(s_k(1)) x_j(0)$ である. ここで, f' はシグモイド関数 f の微分 $\partial f / \partial s_i$ である.

また, $\partial \delta L_m(t) / \partial w_{ij}$ は, (11)式より

$$\frac{\partial \delta L_m(t)}{\partial w_{ij}} = \frac{1}{|L(t)|^2} \left[\frac{\partial L_m(t)}{\partial w_{ij}} |L(t)| - \frac{\partial |L(t)|}{\partial w_{ij}} L_m(t) \right]. \quad (24)$$

である. 初期値 $\partial \delta L_m(0) / \partial w_{ij}$ は, $\delta L_m(0)$ が w_{ij} によらないので $\partial \delta L_m(0) / \partial w_{ij} = 0$ とおくと, (21)式より $\partial L(1) / \partial w_{ij}$ が計算できる. これらより(24)式を用いて $\partial L_m(1) / \partial w_{ij}$ が計算できるから, 以下同

様に $\partial L(t)/\partial w_{ij}$ が計算でき $\partial \lambda/\partial w_{ij}$ が求まり、最終的に δw_{ij} が計算できる。

RNNの最大リアプノフ指数は $\lambda = -1.533$ であった。

5. シミュレーション結果

結合係数をランダムに初期化した $N = 4$ の完全結合型RNNを、任意の目標値に制御することを試みる。ここでは、目標値を $\lambda^{obj1} = 1.0$ と、 $\lambda_1^{obj2} = -8.0$ とし、 $s_o = 0.2$ 、 $T = 100$ とした。なお、シミュレーションにおいてランダムに初期化されたRNNの最大リアプノフ指数は $\lambda = -1.533$ であった。

5.1 提案法の制御性能の検証

提案法の計算式をそのまま用いてシミュレーションを行うと、(23)式、(24)式の値が発散してしまうことが原因で $\delta w_{ij} \rightarrow \infty$ となって計算が行えない場合がある。その対策として $|\delta w_{ij}|$ の上限を適当な値に制限している。 $(\lambda^{obj1}$ の場合:1.0, λ_1^{obj2} の場合:0.01)

しかし、 $\lambda^{obj} > 0$ の場合は、ほとんどの結合係数 w_{ij} で変数量 δw_{ij} の値が発散してしまい制御のための適切な変数量 δw_{ij} が得られず制御に失敗する場合がある(Fig.2)。この原因は、 $\lambda^{obj} > 0$ の領域でネットワークダイナミクスがカオス的なために

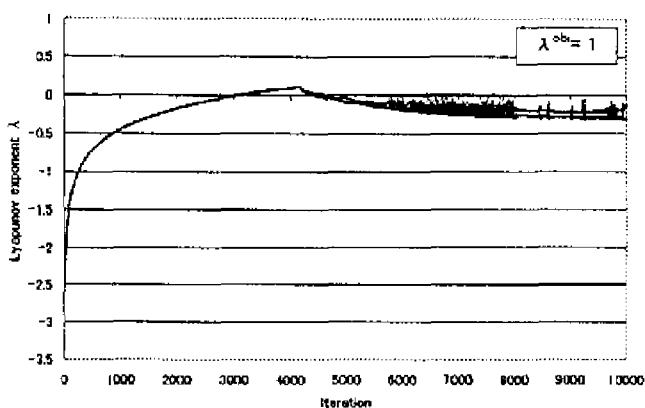


Fig. 2 The Lyapunov exponent λ as a function of the iteration by the proposed control method.

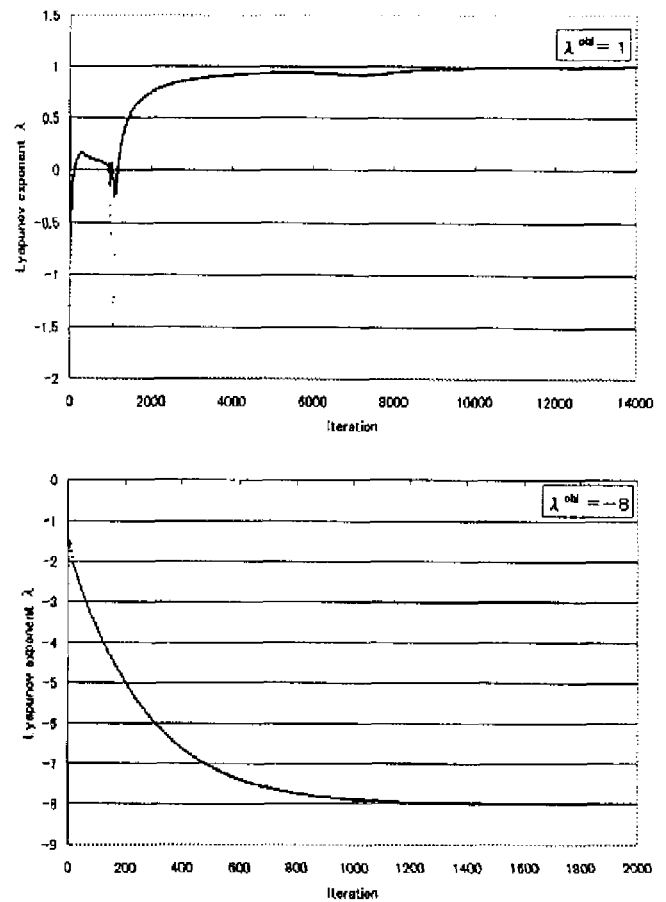


Fig. 3 The Lyapunov exponent λ as a function of the iteration by the proposed control method.

w_{ij} の微小な改変が大きな変化を与えるためと考えられる。

その対策として、本稿では(23)式と(24)式の値を適当な範囲の値に制限した。 $-1 \sim +1$ の範囲に制限した場合のシミュレーション結果をFig.3に示す。この結果より、最大リアプノフ指数の制御に成功していることがわかる。

以上のように、提案法による制御では、目標とする最大リアプノフ指数の値によって、RNNの初期パラメータや計算アルゴリズムに何らかの調整を必要とする場合がある。

6. まとめ

著者等がこれまで提案していた手法によって、RNNのネットワークダイナミクスの複雑さ(最大リアプノフ指数)を目標の値に制御できることを、シミュレーションにて検証した。その結果、提案

法をそのまま用いると、計算が発散してしまう場合があるなどの問題点を確認し、その改善策を示した。

目標とする最大リアプノフ指数が違うと、提案法では、改変量計算に用いるアルゴリズムや定数の値を適切に補正する必要がある。このことから、RNNのパラメータ空間上の λ 曲面は、非常に複雑であり、最大リアプノフ指数の値によって大きく変化していると予想される。したがって、より実用に適した最大リアプノフ指数制御法の構築のためにもRNNのパラメータ空間上の λ 曲面の解析が必要である。

また、対象空間が複雑な場合に用いられるランダム探索法やシンプレックス法などによる設計法との融合も今後の課題である。

参考文献

- 1) 合原一幸: カオス, サイエンス社(1990)
- 2) 中野肇他: ニューロコンピュータ, 第2章, 技術評論社(1989)
- 3) 喜多川, 本間, 阿部: リカレントニューラルネットワークの創発的学習手法, 計測自動制御学会論文集, 33-11, 1093/1098(1997)
- 4) 鎌内, 本間, 阿部: 生物的な認識ダイナミクスをもとに構成した文字認識ニューラルネットワークによる未知パターンの認識, 第10回自律分散システム・シンポジウム資料, 13/16(1998)
- 5) 本間, 酒井, 阿部, 竹田: 神経回路網ダイナミクスの複雑さの制御法, 計測自動制御学会論文集(投稿中)
- 6) A. Wolf, J. B. Swift, H. L. Swinney and J. A. Vastano: Determining Lyapunov Exponents from a time series, *Physica*, 16D, 285/315(1989)